IN THE CLAIMS

Please cancel claims 13-14.

Please amend the claims as follows.

1. (Currently Amended) An apparatus comprising:

at least one processor;

a memory coupled to the at least one processor;

a partition manager residing in the memory and executed by the at least one processor, the partition manager managing allocation of a plurality of hardware resources to a plurality of logical partitions and managing communication between the plurality of logical partitions;

an I/O adapter coupled to the at least one processor;

a device driver for the I/O adapter, the device driver including an interface for performing an I/O operation on a first address of a first length by the I/O adapter, the device driver making a call to the partition manager passing the address of the first length to retrieve a corresponding address of a second length; and

a memory tag mechanism that creates a memory tag of the first length that corresponds to a second address of the second length, wherein the memory tag comprises an identifier that ~~does not represent~~ cannot be mapped to a corresponding location in physical memory;

wherein the partition manager, when the device driver makes the call to the partition manager passing the first address of the first length to retrieve the corresponding address of the second length, detects when the first address is a memory tag, and if so, returns the second address of the second length that corresponds to the memory tag.

2. (Original) The apparatus of claim 1 wherein the partition manager detects when the first address is a memory tag by determining that the first address is in a predefined range of addresses.

3. (Original) The apparatus of claim 1 wherein an address of the second length may be used to initiate a redirected direct memory access operation by the memory tag mechanism creating a memory tag of the first length that corresponds to the address of the second length and passing the corresponding memory tag to the device driver.

4. (Original) The apparatus of claim 1 wherein the first length is 64 bits and the second length is 32 bits.

5. (Currently Amended) A computer-implemented method for ~~accessing~~ a first logical partition to access an I/O adapter allocated to a second logical partition, the method comprising the steps of:

(A) creating a memory tag of a first length that corresponds to a second address of a second length, wherein the memory tag comprises an identifier that ~~does not represent~~ cannot be mapped to a corresponding location in physical memory;

(B) passing the memory tag to a device driver for the I/O adapter;

(C) the device driver passing the memory tag to a partition manager to determine the second address of the second length that corresponds to the memory tag, wherein the partition manager manages allocation of a plurality of hardware resources to the first and second logical partitions and manages communication between the first and second logical partitions;

(D) the partition manager returning to the device driver the second address of the second length that corresponds to the memory tag; and

(E) the device driver accessing the I/O adapter using the second address of the second length that corresponds to the memory tag.

6. (Original) The method of claim 5 further comprising the step of the I/O adapter performing a direct memory access (DMA) operation at a location specified by the second address.

7. (Original) The method of claim 6 wherein the DMA operation comprises a redirected DMA operation.

8. (Original) The method of claim 5 wherein step (D) comprises the step of the partition manager detecting when the first address is a memory tag.

9. (Original) The method of claim 8 wherein the partition manager in step (D) detects when the first address is a memory tag by determining that the first address is in a predefined range of addresses.

10. (Original) The method of claim 5 wherein an address of the second length may be used to initiate a redirected direct memory access operation by creating a memory tag of the first length that corresponds to the address of the second length and passing the corresponding memory tag to the device driver.

11. (Original) The method of claim 5 wherein the first length is 64 bits and the second length is 32 bits.

12. (Currently Amended) A computer readable program product comprising:

(A) a memory tag mechanism that creates a memory tag of a first length that corresponds to a second address of a second length, wherein the memory tag comprises an identifier that ~~does not represent~~ cannot be mapped to a corresponding location in physical memory, the memory tag being passed to a device driver for an I/O adapter, the device driver including an interface for performing an I/O operation on a first address of a first length by the I/O adapter, the device driver making a call to a partition manager passing the address of the first length to retrieve a corresponding address of a second length, the partition manager managing allocation of a plurality of hardware resources to a plurality of logical partitions and managing communication between the plurality of logical partitions, wherein the partition manager detects when the first address is a memory tag, and if so, returns the second address of the second length that corresponds to the memory tag; and

(B) ~~computer readable signal bearing~~ recordable media bearing the memory tag mechanism.

13-14 (Cancelled)

15. (Original) The program product of claim 12 wherein the partition manager detects when the first address is a memory tag by determining that the first address is in a predefined range of addresses.

16. (Original) The program product of claim 12 wherein an address of the second length may be used to initiate a redirected direct memory access operation by the memory tag mechanism creating a memory tag of the first length that corresponds to the address of the second length and passing the corresponding memory tag to the device driver.

17. (Original) The program product of claim 12 wherein the first length is 64 bits and the second length is 32 bits.

Please add the following new claims

18. (New) An apparatus comprising:

at least one processor;

a memory coupled to the at least one processor;

a partition manager residing in the memory and executed by the at least one processor, the partition manager managing communication between first and second logical partitions;

an I/O adapter coupled to the at least one processor and allocated to the first logical partition;

a first device driver for the I/O adapter residing in the first logical partition;

a memory tag mechanism residing in the first logical partition that creates a memory tag of a first length that corresponds to a second address of a second length, wherein the memory tag comprises an identifier that cannot be mapped to a corresponding location in physical memory;

a second device driver for the I/O adapter residing in the second logical partition;

an application residing in the second logical partition that creates a buffer for receiving data from the I/O adapter, the application performing a read operation to the second device driver passing an address of the buffer;

in response to the read operation from the application, the second device driver makes a call to the partition manager passing the address of the buffer, and in response thereto, the partition manager returning a corresponding I/O address for the buffer;

the second device driver passing the I/O address for the buffer to the memory tag mechanism, which generates therefrom a corresponding memory tag for the buffer that cannot be mapped to a corresponding location in physical memory;

the memory tag mechanism making a call to the first device driver, passing the memory tag for the buffer;

the first device driver, in response to the call by the memory tag mechanism, making a call to the partition manager, passing the memory tag for the buffer;

the partition manager returning to the first device driver the I/O address of the buffer;

the first device driver performing a read from the I/O adapter, passing the I/O address of the buffer as a target location for the read; and

writing via direct memory access from the I/O adapter to the I/O address of the buffer.

19. (New) A method for performing a direct memory access operation from an I/O adapter in a first logical partition to an application in a second logical partition, the method comprising the steps of:

the application creating a buffer for receiving data from the I/O adapter;

the application performing a read operation to a second device driver in the second logical partition, passing an address of the buffer;

in response to the read operation from the application, the second device driver making a call to a partition manager passing the address of the buffer;

in response to the call from the second device driver, the partition manager returning a corresponding I/O address for the buffer;

the second device driver passing the I/O address for the buffer to a memory tag mechanism, which generates therefrom a corresponding memory tag for the buffer that cannot be mapped to a corresponding location in physical memory;

the memory tag mechanism making a call to the a device driver, passing the memory tag for the buffer;

the first device driver, in response to the call by the memory tag mechanism, making a call to the partition manager, passing the memory tag for the buffer;

the partition manager, in response to the call by the first device driver, returning to the first device driver the I/O address of the buffer;

the first device driver performing a read from the I/O adapter, passing the I/O address of the buffer as a target location for the read; and

writing via direct memory access from the I/O adapter to the I/O address of the buffer.

20. (New) A computer readable program product comprising:

(A) a memory tag mechanism that creates a memory tag of a first length that corresponds to a second address of a second length, wherein the memory tag comprises an identifier that cannot be mapped to a corresponding location in physical memory, wherein an application residing in a second logical partition creates a buffer for receiving data from an I/O adapter residing in a first logical partition, the application performing a read operation to a second device driver residing in the second logical partition passing an address of the buffer, and in response to the read operation from the application, the second device driver makes a call to a partition manager passing the address of the buffer, and in response thereto, the partition manager returns a corresponding I/O address for the buffer, the second device driver passing the I/O address for the buffer to the memory tag mechanism in the first logical partition, the memory tag mechanism generating from the I/O address for the buffer a corresponding memory tag for the buffer that cannot be mapped to a corresponding location in physical memory, the memory tag mechanism making a call to a first device driver in the first logical partition passing the memory tag for the buffer, the first device driver, in response to the call by the memory tag mechanism, making a call to the partition manager, passing the memory tag for the buffer, the partition manager returning to the first device driver the I/O address of the buffer, the first device driver performing a read from the I/O adapter, passing the I/O address of the buffer as a target location for the read, and writing via direct memory access from the I/O adapter to the I/O address of the buffer; and

(B) recordable media bearing the memory tag mechanism.